Fetal Health Classification Model

DSC 630- Final Project
Evelyn Cates

Executive Summary

Becoming a parent is a life changing experience filled with drastic emotions. From the beginning of pregnancy there are constant examinations to monitor the health of the mother and baby. One of those examinations is Cardiotocography (CTG) that classifies fetal health into Normal, Suspect or Pathological and brings attention to any issues that are occurring.

For this project a dataset was gathered from Kaggle and used to understand what factors contribute the most to fetal health being classified as Pathological. Several different models were implemented to see which models would yield the best results. K Nearest Neighbors became the leading model with a higher accuracy performance over the other two models. By implementing this model physicians can use CTG data to lower prediction errors.

Technical Analysis

Abstract

The purpose of this paper is to create a predictive model using CTG (Cardiotocography) data that will help determine fetal health and lead to less prediction errors. Every pregnancy is different; and conditions are constantly changing, what applies to one person might not be useful to another. Therefore, with this project a classification model was built to classify fetal health to prevent maternal and child death. There were several models used to achieve the best results and K Nearest Neighbors was the leading model with the highest accuracy of 88%.

Background

Throughout pregnancy different tests are predatorially performed to examine the health of the mother and baby, one of those assessments is Cardiotocography (CTG). CTG is usually done during the third trimester of pregnancy where the fetal heart rate and urine contractions are monitored (Potter). CTG provides on the well-being of the fetus and detection of any fetal distress along with maternal health. There is a structure to read CTG along with a helpful acronym DR C BRAVADO. The acronym is broken down to:

- DR: Define Risk
- C: Contractions
- Bra: Baseline Rate
- V: Variability
- A: Accelerations:
- D: Decelerations
- O: Overall impression

For the CTG to assess properly, the first step is to define what the risk factors. Maternally, some of the medical illness is: Gestational diabetes, Hypertension and Asthma. For the obstetric complications it could be multiple gestation, post- date gestation, intrauterine growth restriction, premature rupture of membranes, congenital malformations, pre-eclampsia, oxytocin induction/ augmentation of labor. The following are terminologies that played a significant role in the creation of the model.

Prolonged deceleration: deceleration lasting more than 3 minutes.

Variability: the result of interaction between the nervous system, chemoreceptors, baroreceptors, and cardiac responsiveness. Abnormal variability is considered less than 5 bpm for more than 50 mins, or more than 25 bpm for 25 mins.

These are some of the CTG factors used to determine of the pregnancy is normal or not.

Problem Statement

Constant monitoring is needed throughout pregnancy to determine the health of the mother and baby. By conducting an analysis on fetal health any potential risks can be assessed, monitored, and acted on if necessary. Creating a predictive model would solve the problem of taking necessary measures in the right amount of time.

Methods

Data Exploration

The first step of the project was to start by doing an Exploratory Data Analysis. This helped me understand the information available and which direction was to go with the model such as feature engineering. This was a clean dataset with no missing values. There are 8 rows, 22 columns in total.

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 2126 entries, 0 to 2125
Data columns (total 22 columns):
 #   Column                                                  Non-Null Count  Dtype
---  ------                                                  --------------  -----
 0   baseline value                                          2126 non-null   float64
 1   accelerations                                           2126 non-null   float64
 2   fetal_movement                                          2126 non-null   float64
 3   uterine_contractions                                    2126 non-null   float64
 4   light_decelerations                                     2126 non-null   float64
 5   severe_decelerations                                    2126 non-null   float64
 6   prolongued_decelerations                                2126 non-null   float64
 7   abnormal_short_term_variability                         2126 non-null   float64
 8   mean_value_of_short_term_variability                    2126 non-null   float64
 9   percentage_of_time_with_abnormal_long_term_variability  2126 non-null   float64
 10  mean_value_of_long_term_variability                     2126 non-null   float64
 11  histogram_width                                         2126 non-null   float64
 12  histogram_min                                           2126 non-null   float64
 13  histogram_max                                           2126 non-null   float64
 14  histogram_number_of_peaks                               2126 non-null   float64
 15  histogram_number_of_zeroes                              2126 non-null   float64
 16  histogram_mode                                          2126 non-null   float64
 17  histogram_mean                                          2126 non-null   float64
 18  histogram_median                                        2126 non-null   float64
 19  histogram_variance                                      2126 non-null   float64
 20  histogram_tendency                                      2126 non-null   float64
 21  fetal_health                                            2126 non-null   float64
dtypes: float64(22)
```

Figure 1: Data frame description

For this project, the fetal health column shows the baby's health. It is broken down into three categories Normal (1), Suspect (2) or Pathological (3). Normal fetal health is by far leading which is great news to start and then Suspect and Pathological following.

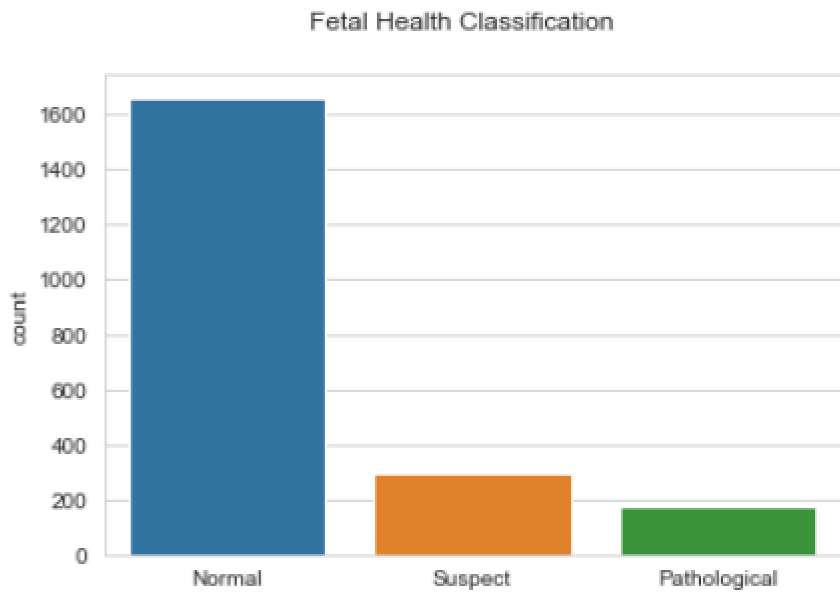
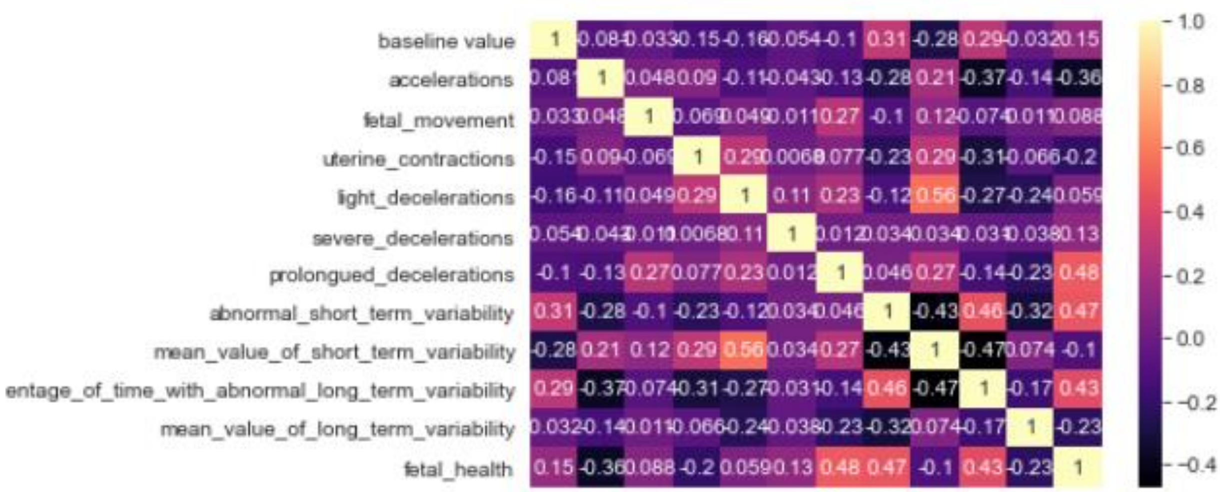


Figure 2: Fetal Health Count Value

There are additional columns that show histogram width, histogram min, histogram max. I went ahead and deleted these columns because I did not think they were necessary to create a model and then made a new data frame. After, I created a correlation heatmap to see which factors contributed to fetal health being categorized as Normal, Suspect or Pathological.

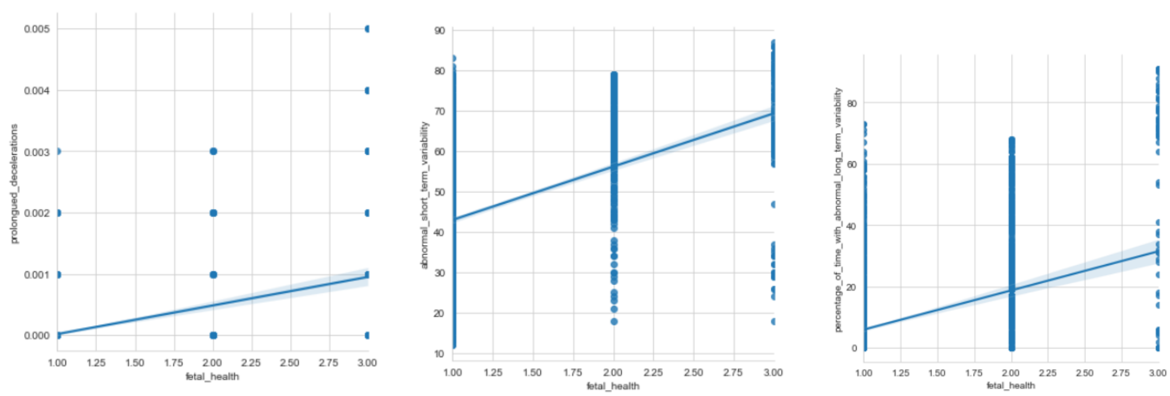


The factors with the highest correlation were:

prolonged decelerations, abnormal short-term variability, and percentage of time with abnormal long-term variability.

	fetal_health
fetal_health	1.000000
prolongued_decelerations	0.484859
abnormal_short_term_variability	0.471191
percentage_of_time_with_abnormal_long_term_variability	0.426146
baseline value	0.148151
severe_decelerations	0.131934
fetal_movement	0.088010
light_decelerations	0.058870
mean_value_of_short_term_variability	-0.103382
uterine_contractions	-0.204894
mean_value_of_long_term_variability	-0.226797
accelerations	-0.364066

The EDA process shaped the model creation because it determined which variables would be the most beneficial. Before creating the model with these variables, I wanted to further see if these variables correlate. I created a linear regression model where each of these variables show a strong correlation and will base the model of these variables.



Building Predictive Models

Since fetal health data is a binary, I choose three models that work well with this type of data, Logistic Regression, K Nearest Neighbors and Random Forest Classifier. It was then split into 70% train set and 30% test set.

```
#splitting the model into test and train sets

X_train, X_test, y_train, y_test = train_test_split(X, y, test_size = 0.3, random_state = 0)

X_train.shape, X_test.shape, y_train.shape, y_test.shape
```

```
((1488, 3), (638, 3), (1488,), (638,))
```

Before applying it to a model, there were preprocessing steps that took place. I standardized the data using StandardScaler.

```
#Preprocessing

scaleX = StandardScaler()
X_train = scaleX.fit_transform(X_train)
X_test = scaleX.transform(X_test)
```

This removed the mean and scaled each feature and variable to a unit variance. In this dataset there were no outliers which would not influence the results of StandardScaler. The data is now ready, and the models can be created.

Results

Model Evaluation - Classification Report and Confusion Matrix

Logistic Regression resulted in an accuracy of 80% macro average and 85% weighted average for precision. Recall accuracy was poor with 65% macro avgerage, but weighted average was 86%.

	precision	recall	f1-score	support
1.0	0.88	0.96	0.92	494
2.0	0.59	0.43	0.50	86
3.0	0.92	0.57	0.70	58
accuracy			0.86	638
macro avg	0.80	0.65	0.71	638
weighted avg	0.85	0.86	0.84	638

KNN had better accuracy results than the LR model with 85% macro avg and 88% weighted average. Recall scores were significantly better with 76% and 89% averages.

	precision	recall	f1-score	support
1.0	0.91	0.96	0.93	494
2.0	0.67	0.59	0.63	86
3.0	0.95	0.72	0.82	58
accuracy			0.89	638
macro avg	0.85	0.76	0.80	638
weighted avg	0.88	0.89	0.88	638

Lastly, Random Forest Classifier received similar accuracy results with 83% macro average and 88% weighted average.

	precision	recall	f1-score	support
1.0	0.91	0.95	0.93	494
2.0	0.72	0.57	0.64	86
3.0	0.85	0.81	0.83	58
accuracy			0.89	638
macro avg	0.83	0.78	0.80	638
weighted avg	0.88	0.89	0.88	638

Additionally, I created a confusion matrix as a second evaluation of all three models. All three models performed well. Overall, KNN performed slightly better and choose as the final model.

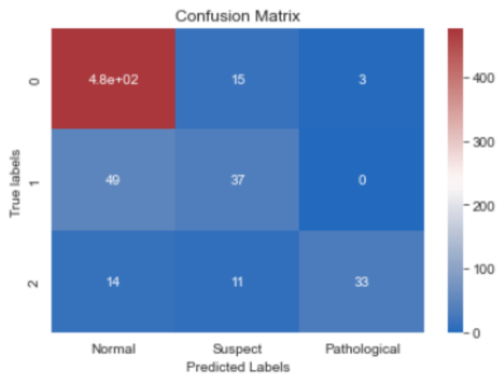


Figure 3 Logistic Regression Confusion Matrix

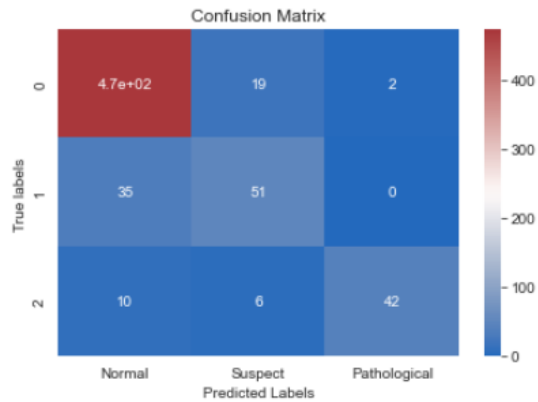


Figure 4 KNN Confusion Matrix

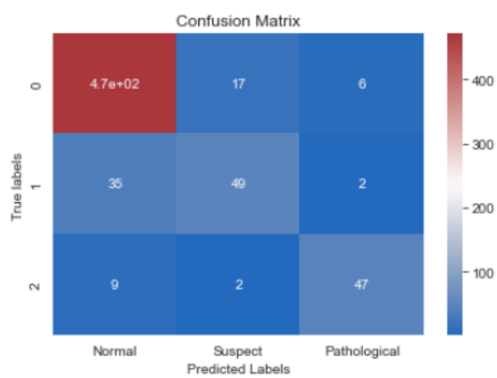


Figure 5 RFC Confusion Matrix

Conclusion

When I first started the project, I originally thought the Logistic Regress would have the best performance because of feature similarity, and it is used in both classification and regression models. It received good results but not it was not the best model. I was surprised with how well results were of the Random Forest Classifier. RFC uses ensemble algorithms that combine more than one algorithm of the same or different kind for classifying objects and in this model led to the best performance and will continue to use in the future. This was also a great learning lesson when it came to selecting different models to build because the model, we think would yield the highest results might not be accurate.

On a personal note, this was an exciting project to work on as a new mom and Data Science student. I learned about CTG and the types of factors determining risks during pregnancy while expanding my skills as a Data Science student.

Acknowledgements

I would like to thank my husbands for helping me stay focused throughout this project and taking over all the baby duties. I would also like to thank my classmates and Professor Werner for your feedback that helped make this project better.

Resources

Potter, L. (2021). How to Read a CTG | CTG Interpretation | Geeky Medics. Retrieved 4 March 2021, from https://geekymedics.com/how-to-read-a-ctg/

Dataset from Kaggle: https://www.kaggle.com/andrewmvd/fetal-health-classification and notebook https://www.kaggle.com/vidyashivakumar/fetal-health-classifier-92-accuracy#EDA